boingboing.net

The coming war on general-purpose computing / Boing Boing

Cory Doctorow

23-30 minutes

By Cory Doctorow - Share this article

This article is based on a keynote speech to the Chaos Computer Congress in Berlin, Dec. 2011.

General-purpose computers are astounding. They're so astounding that our society still struggles to come to grips with them, what they're for, how to accommodate them, and how to cope with them. This brings us back to something you might be sick of reading about: *copyright*.

But bear with me, because this is about something more important. The shape of the copyright wars clues us into an upcoming fight over the destiny of the general-purpose computer itself.

In the beginning, we had packaged software and we had sneakernet. We had floppy disks in ziplock bags, in cardboard boxes, hung on pegs in shops, and sold like candy bars and magazines. They were eminently susceptible to duplication,

were duplicated quickly, and widely, and this was to the great chagrin of people who made and sold software.

Enter *Digital Rights Management* in its most primitive forms: let's call it DRM 0.96. They introduced physical indicia which the software checked for—deliberate damage, dongles, hidden sectors—and challenge-response protocols that required possession of large, unwieldy manuals that were difficult to copy.

These failed for two reasons. First, they were commercially unpopular, because they reduced the usefulness of the software to the legitimate purchasers. Honest buyers resented the non-functionality of their backups, they hated the loss of scarce ports to the authentication dongles, and they chafed at the inconvenience of having to lug around large manuals when they wanted to run their software. Second, these didn't stop pirates, who found it trivial to patch the software and bypass authentication. People who took the software without paying for it were untouched.

Typically, the way this happened is a programmer, with possession of technology and expertise of equivalent sophistication to the software vendor itself, would reverse-engineer the software and circulate cracked versions. While this sounds highly specialized, it really wasn't. Figuring out what recalcitrant programs were doing and routing around media defects were core skills for computer programmers, especially in the era of fragile floppy disks and the rough-and-

ready early days of software development. Anti-copying strategies only became more fraught as networks spread; once we had bulletin boards, online services, USENET newsgroups and mailing lists, the expertise of people who figured out how to defeat these authentication systems could be packaged up in software as little crack files. As network capacity increased, the cracked disk images or executables themselves could be spread on their own.

This gave us DRM 1.0. By 1996, it became clear to everyone in the halls of power that there was something important about to happen. We were about to have an information economy, whatever the Hell that was. They assumed it meant an economy where we bought and sold information. Information technology improves efficiency, so imagine the markets that an information economy would have! You could buy a book for a day, you could sell the right to watch the movie for a Euro, and then you could rent out the pause button for a penny per second. You could sell movies for one price in one country, at another price in another, and so on. The fantasies of those days were like a boring science fiction adaptation of the Old Testament Book of Numbers, a tedious enumeration of every permutation of things people do with information—and what might be charged for each.

Unfortunately for them, none of this would be possible unless they could control how people use their computers and the files we transfer to them. After all, it was easy to talk about selling someone a tune to download to their MP3 player, but

not so easy to talk about the the right to move music from the player to another device. But how the Hell could you stop that once you'd given them the file? In order to do so, you needed to figure out how to stop computers from running certain programs and inspecting certain files and processes. For example, you could encrypt the file, and then require the user to run a program that only unlocked the file under certain circumstances.

But, as they say on the Internet, *now you have two problems*.

You must now also stop the user from saving the file while it's unencrypted—which must happen eventually— and you must stop the user from figuring out where the unlocking program stores its keys, enabling them to permanently decrypt the media and ditch the stupid player app entirely.

Now you have *three* problems: you must stop the users who figure out how to decrypt from sharing it with other users. Now you've got *four* problems, because you must stop the users who figure out how to extract secrets from unlocking programs from telling other users how to do it too. And now you've got *five* problems, because you must stop users who figure out how to extract these secrets from telling other users what the secrets were!

That's a lot of problems. But by 1996, we had a solution. We had the WIPO Copyright Treaty, passed by the United Nations World Intellectual Property Organization. This created laws that made it illegal to extract secrets from unlocking programs,

and it created laws that made it illegal to extract media (such as songs and movies) from the unlocking programs while they were running. It created laws that made it illegal to tell people how to extract secrets from unlocking programs, and it created laws that made it illegal to host copyrighted works or the secrets. It also established a handy streamlined process that let you remove stuff from the Internet without having to screw around with lawyers, and judges, and all that crap.

And with that, illegal copying ended forever, the information economy blossomed into a beautiful flower that brought prosperity to the whole wide world; as they say on the aircraft carriers, “Mission Accomplished".

That's not how the story ends, of course, because pretty much anyone who understood computers and networks understood that these laws would create more problems than they could possibly solve. After all, these laws made it illegal to look inside your computer when it was running certain programs. They made it illegal to tell people what you found when you looked inside your computer, and they made it easy to censor material on the internet without having to prove that anything wrong had happened.

In short, they made unrealistic demands on reality and reality did not oblige them. Copying only got *easier* following the passage of these laws—copying will only ever *get* easier. Right now is as hard as copying will get. Your grandchildren will turn to you and say “Tell me again, Grandpa, about when it

was hard to copy things in 2012, when you couldn't get a drive the size of your fingernail that could hold every song ever recorded, every movie ever made, every word ever spoken, every picture ever taken, everything, and transfer it in such a short period of time you didn't even notice it was doing it."

Reality asserts itself. Like the nursery rhyme lady who swallows a spider to catch a fly, and has to swallow a bird to catch the spider, and a cat to catch the bird, so must these regulations, which have broad general appeal but are disastrous in their implementation. Each regulation begets a new one, aimed at shoring up its own failures.

It's tempting to stop the story here and conclude that the problem is that lawmakers are either clueless or evil, or possibly evilly clueless. This is not a very satisfying place to go, because it's fundamentally a counsel of despair; it suggests that our problems cannot be solved for so long as stupidity and evilness are present in the halls of power, which is to say they will never be solved. But I have another theory about what's happened.

It's not that regulators don't understand information technology, because it should be possible to be a non-expert and still make a good law. MPs and Congressmen and so on are elected to represent districts and people, not disciplines and issues. We don't have a Member of Parliament for biochemistry, and we don't have a Senator from the great state of urban planning. And yet those people who are experts in

policy and politics, not technical disciplines, still manage to pass good rules that make sense. That's because government relies on heuristics: rules of thumb about how to balance expert input from different sides of an issue.

Unfortunately, information technology confounds these heuristics—it kicks the *crap* out of them—in one important way.

The important tests of whether or not a regulation is fit for a purpose are first whether it will work, and second whether or not it will, in the course of doing its work, *have effects on everything else*. If I wanted Congress, Parliament, or the E.U. to regulate a wheel, it's unlikely I'd succeed. If I turned up, pointed out that bank robbers always make their escape on wheeled vehicles, and asked, “Can't we do something about this?", the answer would be “No". This is because we don't know how to make a wheel that is still generally useful for legitimate wheel applications, but useless to bad guys. We can all see that the general benefits of wheels are so profound that we'd be foolish to risk changing them in a foolish errand to stop bank robberies. Even if there were an epidemic of bank robberies—even if society were on the verge of collapse thanks to bank robberies—no-one would think that wheels were the right place to start solving our problems.

However, if I were to show up in that same body to say that I had absolute proof that hands-free phones were making cars dangerous, and I requested a law prohibiting hands-free

phones in cars, the regulator might say “Yeah, I'd take your point, we'd do that."

We might disagree about whether or not this is a good idea, or whether or not my evidence made sense, but very few of us would say that once you take the hands-free phones out of the car, they *stop being cars*.

We understand that cars remain cars even if we remove features from them. Cars are special-purpose, at least in comparison to wheels, and all that the addition of a hands-free phone does is add one more feature to an already-specialized technology. There's a heuristic for this: special-purpose technologies are complex, and you can remove features from them without doing fundamental, disfiguring violence to their underlying utility.

This rule of thumb serves regulators well, by and large, but it is rendered null and void by the general-purpose computer and the general-purpose network—the PC and the Internet. If you think of computer software as a feature, a computer with spreadsheets running on it has a spreadsheet feature, and one that's running World of Warcraft has an MMORPG feature. The heuristic would lead you to think that a computer unable to run spreadsheets or games would be no more of an attack on computing than a ban on car-phones would be an attack on cars.

And, if you think of protocols and websites as features of the network, then saying “fix the Internet so that it doesn't run

BitTorrent", or “fix the Internet so that thepiratebay.org no longer resolves," sounds a lot like “change the sound of busy signals," or “take that pizzeria on the corner off the phone network," and not like an attack on the fundamental principles of internetworking.

The rule of thumb works for cars, for houses, and for every other substantial area of technological regulation. Not realizing that it fails for the Internet does not make you evil, and it does not make you an ignoramus. It just makes you part of that vast majority of the world, for whom ideas like Turing completeness and end-to-end are meaningless.

So, our regulators go off, they blithely pass these laws, and they become part of the reality of our technological world. There are, suddenly, numbers that we aren't allowed to write down on the Internet, programs we're not allowed to publish, and all it takes to make legitimate material disappear from the Internet is the mere accusation of copyright infringement. It fails to attain the goal of the regulation, because it doesn't stop people from violating copyright, but it bears a kind of superficial resemblance to copyright *enforcement*—it satisfies the security syllogism: “something must be done, I am doing something, something has been done." As a result, any failures that arise can be blamed on the idea that the regulation doesn't go far enough, rather than the idea that it was flawed from the outset.

This kind of superficial resemblance and underlying

divergence happens in other engineering contexts. I've a friend, who was once a senior executive at a big consumer packaged goods company, who told me what happened when the marketing department told the engineers that they'd thought up a great idea for detergent: from now on, they were going to make detergent that made your clothes newer every time you washed them!

After the engineers had tried unsuccessfully to convey the concept of entropy to the marketing department, they arrived at another solution: they'd develop a detergent that used enzymes that attacked loose fiber ends, the kind that you get with broken fibers that make your clothes look old. So every time you washed your clothes in the detergent, they would look newer. Unfortunately, that was because the detergent was digesting your clothes. Using it would literally cause your clothes to *dissolve in the washing machine.*

This was, needless to say, the opposite of making clothes newer. Instead, you were artificially aging them every time you washed them, and as the user, the more you deployed the “solution", the more drastic your measures had to be to keep your clothes up to date. Eventually, you would have to buy new clothes because the old ones fell apart.

Today we have marketing departments that say things such as “we don't need computers, we need appliances. Make me a computer that doesn't run every program, just a program that does this specialized task, like streaming audio, or routing

packets, or playing Xbox games, and make sure it doesn't run programs that I haven't authorized that might undermine our profits."

On the surface, this seems like a reasonable idea: a program that does one specialized task. After all, we can put an electric motor in a blender, and we can install a motor in a dishwasher, and we don't worry if it's still possible to run a dishwashing program in a blender. But that's not what we do when we turn a computer into an appliance. We're not making a computer that runs only the “appliance" app; we're taking a computer that can run every program, then using a combination of rootkits, spyware, and code-signing to prevent the user from knowing which processes are running, from installing her own software, and from terminating processes that she doesn't want. In other words, an appliance is not a stripped-down computer—it is a fully functional computer with spyware on it out of the box.

We don't know how to build a general-purpose computer that is capable of running any program *except* for some program that we don't like, is prohibited by law, or which loses us money. The closest approximation that we have to this is a computer with spyware: a computer on which remote parties set policies without the computer user's knowledge, or over the objection of the computer's owner. Digital rights management always converges on malware.

In one famous incident—a gift to people who share this

hypothesis—Sony loaded covert rootkit installers on 6 million audio CDs, which secretly executed programs that watched for attempts to read the sound files on CDs and terminated them. It also hid the rootkit's existence by causing the computer operating system's kernel to lie about which processes were running, and which files were present on the drive. But that's not the only example. Nintendo's 3DS opportunistically updates its firmware, and does an integrity check to make sure that you haven't altered the old firmware in any way. If it detects signs of tampering, it turns itself into a brick.

Human rights activists have raised alarms over U-EFI, the new PC bootloader, which restricts your computer so it only runs “signed" operating systems, noting that repressive governments will likely withhold signatures from operating systems unless they allow for covert surveillance operations.

On the network side, attempts to make a network that can't be used for copyright infringement always converge with the surveillance measures that we know from repressive governments. Consider SOPA, the U.S. Stop Online Piracy Act, which bans innocuous tools such as DNSSec—a security suite that authenticates domain name information— because they might be used to defeat DNS blocking measures. It blocks Tor, an online anonymity tool sponsored by the U.S. Naval Research Laboratory and used by dissidents in oppressive regimes, because it can be used to circumvent IP blocking measures.

In fact, the Motion Picture Association of America, a SOPA proponent, circulated a memo citing research that SOPA might work *because* it uses the same measures as are used in Syria, China, and Uzbekistan. It argued that because these measures are effective in those countries, they would work in America, too!

It may seem like SOPA is the endgame in a long fight over copyright and the Internet, and it may seem that if we defeat SOPA, we'll be well on our way to securing the freedom of PCs and networks. But as I said at the beginning of this talk, this *isn't* about copyright.

The copyright wars are just the beta version of a long coming war on computation. The entertainment industry is just the first belligerents to take up arms, and we tend to think of them as particularly successful. After all, here is SOPA, trembling on the verge of passage, ready to break the Internet on a fundamental level— all in the name of preserving Top 40 music, reality TV shows, and Ashton Kutcher movies.

But the reality is that copyright legislation gets as far as it does precisely because it's not taken seriously by politicians. This is why, on one hand, Canada has had Parliament after Parliament introduce one awful copyright bill after another, but on the other hand, Parliament after Parliament has failed to actually vote on each bill. It's why SOPA, a bill composed of *pure stupid* and pieced together molecule-by-molecule into a kind of “Stupidite 250" normally only found in the heart of

newborn stars, had its rushed-through SOPA hearings adjourned midway through the Christmas break: so that lawmakers could get into a vicious national debate over an *important* issue, unemployment insurance.

It's why the World Intellectual Property Organization is gulled time and again into enacting crazed, pig-ignorant copyright proposals: because when the nations of the world send their U.N. missions to Geneva, they send water experts, not copyright experts. They send health experts, not copyright experts. They send agriculture experts, not copyright experts, because copyright is just not as important.

Canada's Parliament didn't vote on its copyright bills because, of all the things that Canada needs to do, fixing copyright ranks well below health emergencies on First Nations reservations, exploiting the oil patch in Alberta, interceding in sectarian resentments among French- and English-speakers, solving resources crises in the nation's fisheries, and a thousand other issues. The triviality of copyright tells you that when other sectors of the economy start to evince concerns about the Internet and the PC, copyright will be revealed for a minor skirmish—not a war.

Why might other sectors come to nurse grudges against computers in the way the entertainment business already has? The world we live in today is *made* of computers. We don't have cars anymore; we have computers we ride in. We don't have airplanes anymore; we have flying Solaris boxes

attached to bucketfuls of industrial control systems. A 3D printer is not a device, it's a peripheral, and it only works connected to a computer. A radio is no longer a crystal: it's a general-purpose computer, running software. The grievances that arise from unauthorized copies of *Snooki's Confessions of a Guidette* are trivial when compared to the calls-to-action that our computer-embroidered reality will soon create.

Consider radio. Radio regulation until today was based on the idea that the properties of a radio are fixed at the time of manufacture, and can't be easily altered. You can't just flip a switch on your baby monitor and interfere with other signals. But powerful software-defined radios (SDRs) can change from baby monitor to emergency services dispatcher or air traffic controller, just by loading and executing different software. This is why the Federal Communications Commission (FCC) considered what would happen when we put SDRs in the field, and asked for comment on whether it should mandate that all software-defined radios should be embedded in “trusted computing" machines. Ultimately, the question is whether every PC should be locked, so that their programs could be strictly regulated by central authorities.

Even this is a shadow of what is to come. After all, this was the year in which we saw the debut of open source shape files for converting AR-15 rifles to full-automatic. This was the year of crowd-funded open-sourced hardware for genetic sequencing. And while 3D printing will give rise to plenty of trivial complaints, there will be judges in the American South and

mullahs in Iran who will lose their minds over people in their jurisdictions printing out sex toys. The trajectory of 3D printing will raise real grievances, from solid-state meth labs to ceramic knives.

It doesn't take a science fiction writer to understand why regulators might be nervous about the user-modifiable firmware on self-driving cars, or limiting interoperability for aviation controllers, or the kind of thing you could do with bio-scale assemblers and sequencers. Imagine what will happen the day that Monsanto determines that it's *really* important to make sure that computers can't execute programs which cause specialized peripherals to output custom organisms which *literally* eat their lunch.

Regardless of whether you think these are real problems or hysterical fears, they are, nevertheless, the political currency of lobbies and interest groups far more influential than Hollywood and big content. Every one of them will arrive at the same place: “Can't you just make us a general-purpose computer that runs all the programs, except the ones that scare and anger us? Can't you just make us an Internet that transmits any message over any protocol between any two points, unless it upsets us?"

There will be programs that run on general-purpose computers, and peripherals, that will freak even me out. So I can believe that people who advocate for limiting general-purpose computers will find a receptive audience. But just as

we saw with the copyright wars, banning certain instructions, protocols or messages will be wholly ineffective as a means of prevention and remedy. As we saw in the copyright wars, all attempts at controlling PCs will converge on rootkits, and all attempts at controlling the Internet will converge on surveillance and censorship. This stuff matters because we've spent the last decade sending our best players out to fight what we thought was the final boss at the end of the game, but it turns out it's just been an end-level guardian. The stakes are only going to get higher.

As a member of the Walkman generation, I have made peace with the fact that I will require a hearing aid long before I die. It won't be a hearing aid, though; it will really be a computer. So when I get into a car—a computer that I put my body into—with my hearing aid—a computer I put inside my body—I want to know that these technologies are not designed to keep secrets from me, or to prevent me from terminating processes on them that work against my interests.

Last year, the Lower Merion School District, in a middle-class, affluent suburb of Philadelphia, found itself in a great deal of trouble. It was caught distributing, to its students, rootkitted laptops that allowed remote covert surveillance through the computer's camera and network connection. They photographed students thousands of times, at home and at school, awake and asleep, dressed and naked. Meanwhile, the latest generation of lawful intercept technology can covertly operate cameras, microphones, and GPS tranceivers on PCs,

tablets, and mobile devices.

We haven't lost yet, but we have to win the copyright war first if we want to keep the Internet and the PC free and open. Freedom in the future will require us to have the capacity to monitor our devices and set meaningful policies for them; to examine and terminate the software processes that runs on them; and to maintain them as honest servants to our will, not as traitors and spies working for criminals, thugs, and control freaks.

Note: the comments below are closed. Discuss this post in the BBS forums.